

Transmission system

The invention relates to a transmission system for transmitting an information signal via a plurality of subchannels from a transmitter to a receiver.

The invention further relates to a transmitter for transmitting an information signal via a plurality of subchannels to a receiver, to a receiver for receiving encoded
5 information subsignals via a plurality of subchannels from a transmitter, to a method of transmitting an information signal via a plurality of subchannels to a receiver and to a method of receiving encoded information subsignals via a plurality of subchannels from a transmitter.

10 Such a transmission system is known from European Patent Application EP 0 951 091 A2. In this known transmission system multiple transmit antennas are used to transmit streams of coded symbols (i.e. the encoded information subsignals) that originate from the same data source (i.e. the information signal). At the receiver, these multiple streams are received by means of multiple receive antennas and decoded successively by
15 removing the data streams that have been decoded at earlier stages and by canceling the remaining data streams, due to the multiple receive antennas, via spatial (space-time or space-frequency) interference cancellation. Such a scheme is often referred to as an ordered successive interference cancellation (OSIC) scheme.

20 The capacity (throughput) of the known transmission system is limited.

It is an object of the invention to provide a transmission system according to the preamble having a higher transmission capacity than the known transmission system. This object is achieved in the transmission system according to the invention, said
25 transmission system being characterized in that the transmitter comprises:

- a demultiplexer for demultiplexing the information signal into a plurality of information subsignals in dependence on a throughput of the subchannels as ordered by the receiver;

- an encoder for encoding input symbols of the information subsignals into output symbols such that k input symbols of the k -th information subsignal are encoded with a $k \times m$ -code into m output symbols, $1 \leq k \leq m$, said code having the following properties:
 - all k input symbols and all $m-k$ other output symbols are determinable from any k output symbols, and
 - no $m-l$ other output symbols are determinable from any l output symbols, $l < k$;
 - a multiplexer for multiplexing the output symbols into output information subsignals;
- 10 - a channel encoder for channel encoding the output information subsignals into encoded information subsignals;
 - and means for transmitting each encoded information subsignal via one of the subchannels to the receiver;
 - the receiver comprising:
 - 15 - means for receiving the encoded information subsignals;
 - a channel decoder for successively channel decoding the received encoded information subsignals into channel decoded information subsignals by incorporating decoding information of already channel decoded information subsignals;
 - a demultiplexer for demultiplexing the channel decoded information
 - 20 subsignals into channel decoded symbols;
 - a decoder for decoding the channel decoded symbols into decoded output symbols and for supplying the decoding information regarding the decoded output symbols to the channel decoder;
 - a further multiplexer for multiplexing the decoded output symbols into an
 - 25 output information signal. As will be shown later, by using such a code the transmission system can achieve a maximum throughput.

In an embodiment of the transmission system according to the invention the code is a maximum distance separable (MDS) code. MDS codes are widely known and possess the desired properties that enable the encoding principle according to the invention.

- 30 In another embodiment of the transmission system according to the invention the transmitter further comprises an interleaver coupled between the multiplexer and the channel encoder, the interleaver being arranged for interleaving the output information subsignals, wherein the channel encoder is arranged for encoding the interleaved output information subsignals into the encoded information subsignals. This interleaver is meant to

homogeneously spread the information elements within the stream of the output information elements, so as to efficiently use the information elements known from the previously decoded streams for the decoding of the current stream.

In a further embodiment of the transmission system according to the invention
5 the channel decoder is arranged for decoding a received encoded information subsignal by incorporating decoding information of the most recently channel decoded information subsignal. In such a case, the most recent update on an information element of an information subsignal may incorporate all the available estimates (reliability measures) on the information elements shared through the previously decoded substreams, thereby providing
10 the highest reliabilities of these information elements.

The above object and features of the present invention will be more apparent from the following description of the preferred embodiments with reference to the drawings,
15 wherein:

Figure 1 shows a block diagram of a transmission system 10 according to the invention,

Figure 2 shows a block diagram of a receiver 16 according to the invention,

Figure 3 shows a block diagram of an encoder structure including an MPC
20 encoder 30 for use in a transmitter 12 according to the invention,

Figure 4 shows a block diagram of a decoder structure including an MPC decoder 80 for use in a receiver 16 according to the invention,

Figure 5 shows a block diagram of a prior art transmitter 12,

Figure 6 shows a block diagram of a prior art wireless receiver 16,

25 Figure 7 shows a block diagram of an embodiment of a transmitter 12 according to the invention,

Figure 8 shows a block diagram of an embodiment of a receiver 16 according to the invention,

Figures 9 to 14 show some graphs illustrating the performance of a
30 transmission system 10 according to the invention,

Figures 15 and 16 show a lookup table for use in hard decision decoding of the (3,2,1) MDS code,

Figure 17 shows computation rules for use in MAP decoding of the (3,2,1) MDS code,

Figure 18 shows computation rules for use in soft ML (Max-Log-MAP) decoding of the (3,2,1) MDS code.

In the Figures, identical parts are provided with the same reference numbers.

5

This invention relates to a transmission system 10 as shown in Figure 1 which transmission system 10 makes use of multiple subchannels to deliver information from a transmitter 12 to a receiver 16. In the receiver 16, the signals from different subchannels are decoded successively by decoders 18. The order of decoding is defined at the receiver 16 and is unknown to the transmitter 12. It is assumed that the transmitter 12 is aware of the ordering principle and of some properties of the ordered sequence of subchannels such as statistics (e.g. Rayleigh fading) of the capacities (maximum throughputs) of the ordered subchannels. Alternatively, the transmitter 12 may be aware of the signal to noise (and interference) ratio (SNR/SINR) of the ordered subchannels. However, the order of decoding for each realization of subchannels is random, as seen by the transmitter 12. The present invention is related to a channel coding strategy that makes use of the known properties of the subchannels to enhance the performance of such a multi-channel transmission system 10. This channel coding strategy comprises an encoder and a decoder architecture. The general coding strategy is further applied to enhance the performance of wireless transmission systems that make use of multiple transmit antennas to transmit multiple parallel data streams and multiple receive antennas with an ordered extraction of the transmitted streams at the receiver known as ordered successive interference cancellation (OSIC).

First, we explain the basic idea of the proposed channel coding strategy which allows to reach the maximum achievable throughput (Shannon capacity) of multiple channels with ordered successive decoding. To this end, the equivalent scheme of such a channel will be defined and a fundamental limitation on the throughput of such a scheme will be shown. Next, a forward error correction (FEC) structure that approaches the fundamental throughput bound will be described.

The general case of a system with an arbitrary number m of subchannels, ordered successive decoding of the corresponding data streams at the receiver 16, with a known sequence of ordered capacities, is illustrated by the block diagram in Figure 1. According to this diagram, in the transmitter 12 a set of user bits is encoded by an encoder 8 into m parallel streams that are transmitted over an m input m output channel 14. This channel 14 is represented by a permutation π of the m transmitted streams followed by m

parallel subchannels with the respective capacities (or throughputs) $C_1 \dots C_m$. Precisely, $\pi[n]$ specifies the index of the transmitted stream that is permuted to the n -th subchannel. The set of capacities is known at the receiver 16 and to the transmitter 12 whereas the permutation π is known at the receiver 16 only. The transmitter 12 treats π as a random permutation (specifically, π is assumed uniformly distributed over the set of m possible permutations). At the receiver 16, the m streams are decoded successively so that n -th decoder 18 makes use of the knowledge of the user (input) bits recovered by the previous $(n-1)$ decoders 18, $1 < n \leq m$.

The overall throughput (channel capacity) of such system is limited by the sum of throughputs of individual subchannels, i.e.

$$C_x \leq C_1 + \dots + C_m. \quad (1)$$

This capacity may be only reached when the rate at the n -th output of the permutation π is matched to the respective capacity C_n , $1 \leq n \leq m$. This latter condition along with the fact that π is unknown at the transmitter 12 has a number of implications for an optimal encoding rule. First of all, each transmitted data stream should carry data at rate $R_1 = C_1$. If the rate of the k -th stream differs from C_1 , it will not fit the respective throughput/capacity in the event that $\pi[1] = k$. Hence, for a block of N channel uses (complex signal dimensions), the total number of bits within each stream should be $N C_1$. In the same manner, one can see that the conditional rate of any stream upon decoding of any other stream should be $R_{2n} = C_2$.

Indeed, the stream decoded at the second stage will benefit from a partial knowledge of the bits that is provided by the first stream. In order to match the throughput of the second subchannel, this partial knowledge should lead to a rate reduction from C_1 to C_2 .

Consequently, the remaining number of unknown bits in any stream upon decoding of any other stream should equal $N C_2$. This implies that the decoder of the first (uppermost)

subchannel should carry over the knowledge of $N (C_1 - C_2)$ bits to the second subchannel.

By iterating the same argument, it is easy to show that the decoder of the n -th subchannel should carry $N (C_n - C_{n+1})$ additional bits over to the decoders of the following subchannels.

In this case, the remaining number of unknown bits at the $(n+1)$ -th subchannel equals to the difference between the total of $N C_1$ bits carried by a stream at the $(n+1)$ -th subchannel and the total number of bits carried over by the previous decoding stages:

$$NC_1 - N(C_1 - C_2) - \dots - N(C_n - C_{n+1}) = NC_{n+1} \quad (2)$$

This number matches the throughput NC_{n+1} of the $(n+1)$ -th subchannel.

The described flow of decoding stages is represented schematically in Figure 2. This block diagram shows the number of newly decoded user bits at each decoding stage 18 as well as the number of bits carried over to the following stages 18. Clearly, all the bits that are carried over to the decoder 18 of any subchannel from the previous decoding stages 18 should be shared/included with the bits encoded in the data stream which is fed to this subchannel. Furthermore, the relationship between the streams should be fully symmetric w.r.t. any reordering of the streams, since π is unknown at the transmitter 12.

One can notice that the block diagram in Figure 2 implies a non-negative difference between the throughputs of the adjacent stages and therefore

$$C_1 \geq C_2 \geq \dots \geq C_m. \quad (3)$$

This condition is fundamental to reach the equality in (1) for the general system in Figure 1.

In this system, the data rate at the n -th subchannel is a conditional rate $R_{n|1\dots n-1}$ of any transmitted stream given the knowledge of any other $(n-1)$ streams decoded at the preceding stages. On one hand, a sequence of conditional rates is non-increasing. On the other hand, the maximum throughput is only achieved when these rates are matched to the throughputs of the respective subchannels, i.e. $R_{n|1\dots n-1} = C_n$, $1 \leq n \leq m$. Hence the condition (3) is mandatory to achieve the maximum throughput. Whenever this condition does not hold, the maximum throughput of the system in Figure1 will be limited by

$$C_{\Sigma} = \underline{C}_1 + \dots + \underline{C}_m, \quad \text{wherein} \quad \underline{C}_n = \min \{C_1, \dots, C_n\}, \quad 1 \leq n \leq m \quad (4)$$

Note that the sequence $\{\underline{C}_1, \dots, \underline{C}_m\}$ maximizes C_{Σ} from (4) under the constraint (3).

Next, we describe a new encoding scheme that allows for the total throughput specified in (4). This scheme involves a class of codes widely known as maximum distance separable (MDS) codes. An MDS code can be specified by a triple $(n, k, 2^q)$ wherein k is the number of input (information) symbols, n is the number of output (coded) symbols and q specifies the alphabet $GF(2^q)$. This means that input and output symbols belong to the

alphabet $A = \{0, 1, \dots, 2^q - 1\}$. An MDS code is a linear code, hence it may be described by a $k \times n$ generator matrix G with entries from A . A vector of k input symbols $\mathbf{a} = [a_1, \dots, a_k]$ from the alphabet A is encoded into n output symbols $\mathbf{c} = [c_1, \dots, c_n]$ from the same alphabet via

$$\mathbf{c} = \mathbf{a} G \quad (5)$$

wherein multiplications / additions take place in $GF(2^q)$, i.e. the result of standard operations (multiplications & additions) is taken modulo 2^q . The definition given above includes the whole class of linear block codes. The subclass of MDS codes is characterized by such generator matrices G that any arbitrary subset of k symbols from the set of n output symbols of \mathbf{c} yields the set of input symbols \mathbf{a} and therefore the remaining $(n - k)$ output symbols. An important consequence of the above definition of MDS codes is that knowledge of any set of less than k output symbols of \mathbf{c} gives no information about the remaining output symbols.

The last two properties of the MDS codes are essential to design the encoding scheme that matches the decoding flow indicated in Figure 2 and therefore allows to achieve the maximum total throughput given by (4). The construction of such codes is based on MDS codes and will be explained in the following section. Nevertheless, any other code having the same two essential properties as the known MDS codes may equally well be used in the present invention.

Some more facts about MDS codes w.r.t code construction: First of all, we note that a linear code is MDS if and only if its $k \times n$ generator matrix G is such that all $k \times k$ blocks of G are full rank. Such a matrix may be built if $n \leq 2^q + 1$. It is worth mentioning that binary MDS codes (i.e. codes operating with bits, $q = 1$) exist for $n \leq 3$ only. For $n > 3$, one has to make use of $q > 1$. However, a (n, k, q) MDS with $q > 1$ may be adapted to encode a stream of bits. A possible solution consists of the following steps:

- group the stream of input bits into a stream of q -tuples of bits;
- map of these q -tuples to the symbols from alphabet A ;
- encode of the blocks of k symbols according to (5);
- de-mapping of the resulting output symbols from A to q -tuples of bits.

Clearly, the total number of input/output bits of such an encoder should be a multiple of q . Next, we will see that the described four step encoding procedure allows us to

match the decoding flow in Figure 2 and therefore reach maximum throughput with any $q \geq 1$.

Let us show a few simple cases of MDS codes. We make use of these codes in the examples of code construction later in this document.

$$\begin{aligned} (2, 1, 1) \text{ MDS code: } G &= \begin{bmatrix} 1 & 1 \end{bmatrix}, & (3, 1, 1) \text{ MDS code: } G &= \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}, \\ (3, 2, 1) \text{ MDS code: } G &= \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}. \end{aligned} \quad (6)$$

- 5 We finally note that a systematic design of MDS codes may be achieved. One possibility is to use extended Reed-Solomon codes.

Next, we describe the operation of the forward error correction (FEC) structure that enables the full throughput (4). The core engine of this structure is a so-called message-partition code (MPC), based on cascaded MDS codes. Block-diagrams of the entire encoder and decoder are given in Figure 3 and Figure 4 respectively. For sake of simplicity, assume that (3) is satisfied (otherwise $\{C_1, \dots, C_m\}$ should be replaced by $\{\underline{C}_1, \dots, \underline{C}_m\}$). An information signal 21 comprising a set of $N(C_1 + \dots + C_m)$ user bits 21, corresponding to the maximum throughput of $(C_1 + \dots + C_m)$, is fed to the input of the encoder 30. This set is further partitioned/demultiplexed by a demultiplexer 20 into m subsets 33 (or information subsignals 33) with sizes $N(C_1 - C_2)$, $N2(C_2 - C_3)$, \dots , $N(m-1)(C_{m-1} - C_m)$ and NmC_m . The first subset 33 serves as input to an MDS encoder 32 with parameters $(m, 1, \bullet)$. The last parameter (alphabet size) is immaterial for our construction and may be chosen upon the convenience (availability) of an MDS code with appropriate dimensions. Note that each input symbol of a $(m, 1, \bullet)$ MDS code produces m output symbols. The whole block of $N(C_1 - C_2)$ bits will result in m parallel streams 35 of $N(C_1 - C_2)$ output bits. These streams 35 are fed to m multiplexers 34 in Figure 3. In the same way, the k -th subset 33 containing $Nk(C_k - C_{k+1})$ user bits is fed to an MDS encoder 32 with parameters (m, k, \bullet) , $1 \leq k < m$. Such a code encodes groups of k user symbols into respective groups of m output symbols. Again, the whole block of $Nk(C_k - C_{k+1})$ user bits results in m parallel streams 35 of $N(C_k - C_{k+1})$ output bits. These streams 35 are fed to m multiplexers 34, similarly to the outputs of the first MDS encoder 32. Finally, the very last subset 33 of NmC_m is simply partitioned into m parallel streams 35 of NC_m bits, which streams are fed to m multiplexers 34 as indicated in Figure 3. The outputs 31 (or output information subsignals 31) of the

multiplexers 34 represent the m outputs of the MPC encoder 30. It is easy to see that each output 31 carries NC_1 bits. Note that this number matches the required rate of each transmitted data stream.

The output information subsignals 31 should undergo the conventional steps of FEC encoding (represented schematically by the channel encoder blocks 24 in Figure 3) and modulation prior to the transmission. Here, we assume an ideal encoding at the transmitter 12 such that the codebooks are optimally matched to the statistical properties of the transmission channel and a subsequent optimal processing at the receiver 16. The output information subsignals 31 are encoded by the channel encoders 24 into encoded information subsignals 25.

According to the block-diagram in Figure 1, the $\pi[1]$ -th transmitted stream 45 (i.e. the received encoded information subsignal 45) is the first to be processed at the receiver 16. Since any transmitted stream carries NC_1 bits, the corresponding rate matches the throughput C_1 of the first subchannel as indicated in Figure 1. Hence, an optimal channel decoder 46 in Figure 4 restores NC_1 bits 47 (i.e. the channel decoded information subsignal 47) of the first data stream. These bits 47 are fed to the input "input #1" of the MPC decoder 80. This set of bits 47 is partitioned by an upper-most demultiplexer 82 into m streams 95, 97, 99, 101 of sizes $N(C_1 - C_2)$, $N(C_2 - C_3)$, ..., $N(C_{m-1} - C_m)$, NC_m , respectively, inversely to the corresponding multiplexer 34 of the MPC encoder 30 in Figure 3. The first stream 95 contains $N(C_1 - C_2)$ bits that correspond to the $\pi[1]$ -th output of the $(m, 1, \bullet)$ MDS code. These bits 95 are sent to the decoder 84 of the $(m, 1, \bullet)$ MDS code. By definition of $(m, 1, \bullet)$ MDS code, any of its output symbols yields knowledge of the corresponding input as well as the other $(m-1)$ outputs. Thus the upper-most I-MDS decoder 84 restores the subset of $N(C_1 - C_2)$ input bits 117 (i.e. the decoded output symbols 117) of the $(m, 1, \bullet)$ MDS code as well as the remaining $(m-1)$ output streams 83 (i.e. the decoding information 83) of size $N(C_1 - C_2)$. While the subset of input bits 117 is collected by the user bit buffer/multiplexer 86, the $(m-1)$ streams 83 of $N(C_1 - C_2)$ output bits are passed over to the following decoding stages through the output labeled as "link #1". Note that the number of bits passed to the following stages matches the decoding flow diagram in Figure 2. Next, the $\pi[2]$ -th transmitted stream 59 is processed. This stream 59 also carries NC_1 bits of which $N(C_1 - C_2)$ bits 83 are provided by the previous stage of decoding. Note that the remaining

number of bits carried by the $\pi[2]$ -th stream equals $(NC_1 - N(C_1 - C_2)) = NC_2$. This number matches the throughput C_2 of the second subchannel as indicated in Figure 1 so that an optimal decoder 52 in Figure 4 restores all NC_1 bits 47 of the second data stream 59. These bits 47 are fed to the MPC decoder 80 via "input 2" wherein they are

5 partitioned/demultiplexed by the second demultiplexer 82 into m streams of channel decoded symbols, the streams having sizes $N(C_1 - C_2)$, $N(C_2 - C_3)$, \dots $N(C_{m-1} - C_m)$, NC_m . Note that the stream containing $N(C_1 - C_2)$ bits coincides with the set of bits carried over from the previous decoding stage; this set is of no use. The stream 103 that contains $N(C_2 - C_3)$ bits is sent to the I-MDS decoder 84 of the $(m, 2, \bullet)$ MDS code, along with a set

10 97 of $N(C_2 - C_3)$ bits provided by the demultiplexer 82 of the first decoding stage. These two streams 97, 103 provide a pair of outputs for each of $N(C_2 - C_3)$ input blocks of the $(m, 2, \bullet)$ MDS code. By definition of $(m, 2, \bullet)$ MDS code, any pair of its outputs yields knowledge of the corresponding two inputs as well as the other $(m - 1)$ outputs. Thus the corresponding I-MDS decoder 84 restores the subset 119 of $N2(C_2 - C_3)$ input bits as well

15 as the remaining $(m - 1)$ output streams 87 of size $N(C_2 - C_3)$. The subset 119 of $N2(C_2 - C_3)$ input bits is collected in the multiplexer 86. One of the remaining $(m - 1)$ output streams is shared with the $\pi[1]$ -th stream that was recovered at the previous stage. This stream has no further use. The other $(m - 2)$ output streams 87 are passed to the following decoding stages via the output "link #2". Once again, the number of bits passed to

20 the following stages matches the decoding flow diagram in Figure 2. The remaining decoding stages proceed in a similar way.

At the m -th decoding stage, the $\pi[m]$ -th transmitted stream 73 is processed. This stream carries NC_1 bits. The subsets 83, 87, 91 of $N(C_1 - C_2)$, $N(C_2 - C_3)$, \dots $N(C_{m-1} - C_m)$ bits from this stream are provided by the decoding stages 1 through $(m - 1)$

25 via the outputs "link 1" through "link m ", respectively. The remaining number of unknown bits is therefore

$$NC_1 - N(C_1 - C_2) - \dots - N(C_{m-1} - C_m) = NC_m$$

This number matches the throughput C_m of the m -th subchannel as indicated in Figure 1.

Hence an optimal decoder 76 in Figure 4 restores this stream. The resulting NC_1 bits are sent

to the MPC decoder 80 via “input m ”. The following demultiplexer 82 selects NC_m bits 109 that are restored at this stage (since the other $N(C_1 - C_m)$ bits out of NC_1 are provided by the previous decoding stages). Note that these bits are part of the m -th subset of NmC_m user bits in the MPC encoder 30 (see Figure 3). The remaining $(m-1)$ blocks of NC_m user

5 constituting the m -th subset of NmC_m bits in the MPC encoder are coming from the previous decoding stages as they contribute to the previously decoded transmitted streams. These m blocks are grouped to the decoded subset of NmC_m user bits and fed to the multiplexer 86. Check that the total number user bits collected by the multiplexer 86 after the m -th decoding stage equals

$$N(C_1 - C_2) + N2(C_2 - C_3) + \dots + NmC_m = N(C_1 + C_2 + \dots + C_m)$$

10 i.e. the total number of the transmitted user bits. The decoding is now complete.

First, a transmission system 10 having MIMO fading channels with additive Gaussian noise will be discussed. In this context it is assumed that the signal at the output of each subchannel is given by a linear combination of the symbols transmitted in all

15 subchannels and an additive Gaussian noise. In such scenarios, with an appropriate processing at the receiver 16, each subchannel may be regarded as a scalar channel corrupted by a residual inter-subchannel interference and additive Gaussian noise. The commonly used coding strategy consists of a bit-space encoding (FEC encoder), mapping the coded bits into channel symbols and putting these symbols into channel (modulation). The last step depends

20 on the channel properties. Usually, modulation is accomplished in time domain (single carrier systems) or frequency domain (multicarrier systems). In both cases, spreading may be applied so that several channel symbols share several channel uses in time or frequency domain (giving rise to direct sequence or multicarrier spread spectrum transmission, respectively). The choice of channel symbol alphabet (signaling) depends on the desired

25 spectral efficiency and FEC rate. The commonly used signaling schemes are: BPSK, QPSK, 8-PSK and 2^k -QAM with $k \geq 2$. Note that the proposed encoding strategy may be applied for such MIMO fading channels. To this end, one has to specify the set of capacities $C_1 \dots C_m$ that are characterized by the signal to interference plus noise ratios (SINR) within the respective subchannels. In the case of fading channels, the SINR may be unknown to the

30 transmitter. The standard approach is to use outage values of SINR that are chosen, according to the expected statistical properties of the fading, as a lower bound on the actual unknown

SINR for all but a small subset of outage channels. Note that the outage value of SINR/capacity/etc. is the value of SINR/capacity/etc. for which the outage rate is equal to a certain percentage of the cases/time when the actual SINR/capacity/etc. of the system is worse than the outage value.

5 The FEC coding strategies commonly used for channels with Gaussian noise are the standard convolutional codes and, since recently, parallel and serial concatenated interleaved (turbo) codes and low density parity check (LDPC) codes. Although all these codes may be incorporated within the encoders 24 in Figure 3, the efficiency of our approach may depend on the properties of the chosen FEC code. Indeed, the error correction capability
10 of the known user bits at the input of the FEC encoders 24 depends on the span of typical error patterns that involve the known bits. In the case of random-like codes, such as LDPC and turbo codes, typical error patterns span a substantial part of the coded bits. Therefore, every known bit at the input of the encoders 24 is expected to have a global error correction effect over the whole code (unlike a local effect in the case of convolutional codes). It is,
15 therefore, expected that the use of turbo codes, LDPC or similar codes FEC codes will be particularly beneficial for the general MIMO encoding scheme disclosed here. For such FEC codes, the contributions of different MDS codes should be homogeneously mixed. Such a mixing may be achieved by an interleaver (not shown) coupled between the multiplexer 34 and the channel encoder 24 and which interleaver performs a pseudo-random (uniform)
20 interleaving of each output 31 of the MPC encoder 30 prior to the FEC encoding (e.g. by turbo or LDPC codes), see Figure 3. An interleaver may be further optimized subject to a chosen FEC code.

 The general decoding scheme has been described above and is depicted by Figure 4. At the n -th decoding stage, a FEC decoder 46,52,76,128 is activated which makes
25 use of a partial knowledge of the inputs of the corresponding encoder 32. This knowledge is provided by the previous $(n-1)$ decoding stages. A practical implementation of each decoder depends on the type of code and the corresponding decoding procedure. Here we distinguish between soft and hard decision decoders. In hard decision decoding, a FEC decoder generates binary decisions on the input user bits. These decisions on previously
30 decoded bits (information elements) are used at the subsequent decoding stages to limit the choice of possible codewords.

 Example. For convolutional codes, binary decisions on the input bits usually result from a ML sequence detection by Viterbi algorithm. The decisions on the previously decoded bits are then incorporated in the Viterbi algorithm of the successive decoding stages

so that, at every trellis section associated to the previously decoded bit, only those state transitions are considered that correspond to the decoded binary value (thereby reducing the total number of possible transitions by 2 if the conventional Viterbi algorithm is applied).

In soft decision decoding, a FEC decoder generates (soft) real-valued metrics that represent reliability measures of the input bits. Usually, each soft metric is (an approximate) log-likelihood ratio, i.e. the logarithm of the ratio of a posteriori probability of an input bit being 0 versus a posteriori probability of this bit being 1, given the observed signal. Soft metrics are often involved in decoding procedures of concatenated codes and iterative decoding algorithms. The final decision on the input bit is taken according to the sign of such soft metrics. Whenever soft decisions on the input bits are available, the use of soft metrics on the bits provided by the previous decoding stages, instead of binary decisions (or equivalently, very big soft values) ensures generally a better performance.

Example: Soft decision decoding is commonly used to iteratively decode turbo codes and LDPC codes. In the case of turbo codes, soft metrics of the input bits are generated by the so-called soft-input soft-output (SISO) decoders of the component codes. For LDPC codes, the soft metrics result from the so-called belief propagation algorithm. Whenever such FEC codes are used within the MIMO encoding scheme disclosed here, soft metrics of the appropriate bits, obtained at the last iteration of the current stage, are transferred to the subsequent decoding stages. At the subsequent stages, the so-obtained soft metrics should be used as a priori metrics on the appropriate bits or added to the existing a priori metrics if the latter are available at these decoding stages.

Iterative decoding may be applied to improve the performance of the proposed MIMO scheme. Iterative decoding of the scheme in Figure 1 means repeating the whole or any part of the full decoding cycle which consists of successive decoding of m subchannels as described above. In this case, the early decoding stages can make use of hard/soft information on the shared bits obtained at the subsequent stages during the previous iterations. In the case of soft decision decoding, the reliability values to be used at other stages should be computed according to the standard rules for extrinsic information, similarly to turbo decoding, in order to avoid double counting of the same information.

It remains to explain the operation of the MDS decoders. Here again, we will distinguish between hard and soft decision decoding. In the former case, the decoder is driven by binary decisions on a subset of output bits of the respective MDS code in order to find the corresponding input bits as well as the remaining output bits. A large variety of algebraic decoding methods may be used. Since the number of parallel channels is often

small in the situations of interest, the associated decoding complexity is not important. One can use a simple syndrome decoding or even an exhaustive search over all possible codewords of an MDS code. In the case of soft decision decoding, the decoder receives real-valued metrics of a subset of output bits. These metrics reflect reliabilities of the respective bits, as explained above. Based on these metrics, the decoder computes the metrics of the corresponding input bits and the metrics of the remaining output bits. The computation rule is the same as in SISO decoding of turbo codes: the computed metrics of the stand for a posteriori reliabilities of the respective bits which are based on the a priori reliabilities a subset of output bits fed to the decoder. The relationship between the a priori and a posteriori reliabilities depends on the relationship between the corresponding bits. The latter relationship is defined by a generator matrix of the MDS code. The complexity of soft-decision decoding is relatively small when such is m .

The same channel coding principle may be applied to a binary MIMO transmission channel. A possible scenario is transmission of binary messages (data packets) in a network from a transmitter to a receiver via multiple routes (subchannels). Assume that exact reliabilities of different paths (crossover probabilities of the equivalent binary symmetric channels) are not known to the transmitter but the statistical properties (a distribution law of the crossover probability) are known, e.g. from modeling results. One way to deal with the uncertainty of reliabilities per subchannel is to decode these streams in such an order that the reliabilities of the ordered subchannels form a non-increasing sequence. As a matter of fact, variations of each element of an ordered sequence of independent identically distributed random values go to zero as the sequence length goes to infinity. Hence, one can accurately adapt transmission rates to the (quasi-deterministic) throughputs of the ordered subchannels provided that their number is big enough. Although the throughputs of the ordered subchannels may be assumed accurately known at the transmitter, the order of the extraction of the transmitted streams is unknown (unless a feedback channel between the transmitter and the receiver is used to convey channel information). In such a case, the overall MIMO channel falls under the scope of the general scheme in Figure 1. Further implementation may be achieved based the available FEC schemes for binary symmetric channels (i.e. with Reed-Solomon codes and other BCH codes). Yet, it is desirable to adapt the existing coding schemes so as to emphasize the benefit of a partial knowledge of the input bits, according to the proposed channel coding strategy.

The channel coding strategy described above may be applied to increase the throughput of wireless communication systems that exploit multiple transmit and receive

antennas. In such systems, multiple transmit antennas 28 are used to transmit streams of coded symbols that originate from the same data source. At the receiver side, these multiple streams are retrieved and decoded either simultaneously or successively. Simultaneous decoding of different streams yields a very high computational burden. This burden grows exponentially in the total number of bits per channel use transmitted by all antennas 28. Therefore, simultaneous decoding is feasible only for small data rates as compared to the theoretical throughput. Here, we focus on the successive decoding schemes where each data stream is recovered by removing the data streams that have been recovered at earlier stages and by canceling the remaining data streams, due to multiple receive antennas 40, via spatial (space-time or space-frequency) interference cancellation. Specifically, we consider schemes with ordered successive interference cancellation (OSIC).

A baseline system that makes use of the OSIC principle is disclosed in European Patent Application EP 0 951 091 A2. According to this known system, the total number of user bits 21 is partitioned into m symmetric streams 23. Each stream 23 undergoes an identical encoding (by encoders 24), modulation (by modulators 26) and is transmitted by one of m transmit antennas 28. A block diagram of such a transmitter 12 is shown in Figure 5. The receiver 16 exploits M antennas 40 that produce M signal outputs. The receiver 16 applies the OSIC principle which is schematically shown in Figure 6. It is assumed that the transfer function of the MIMO channel is known or accurately estimated at the receiver 16 (e.g. due to the standard training procedure, based on the reference signals sent by the transmitter 12). This MIMO transfer function will be schematically described by an $M \times m$ matrix H whose entry $H_{q,p}$ stands for the transfer function between the p -th transmit antenna 28 and the q -th receive antenna 40. In frequency selective fading, the entries of H are functions representing either time domain or frequency domain characterization of the channel. In non-selective (flat) fading, the entries of H are complex valued.

The received encoded information subsignals are demodulated in demodulators 42. Based on the known H , the receiver 16 extracts the m streams (information subsignals) successively. At the first (leftmost) layer or stage (comprised of a MMSE canceller 44, a decoder 46, an encoder/modulator 48, multipliers 56 and subtractors 60), one of the streams 41 is extracted, by canceling the contributions from the other $(m-1)$ streams 41. Without loss of generality, assume that the index of the stream extracted at the first layer is $\pi[1]$. In the known system, a perfect cancellation of these streams is achieved

due to projecting the vector $H_{\pi[1]} = [H_{1,\pi[1]}, \dots, H_{M,\pi[1]}]^T$ (the superscript (T) stands for a matrix transpose) of channel transfer functions associated to this stream onto the part of the M -dimensional signal space which is orthogonal to the columns of the $M \times (m-1)$ matrix $[H_{1:M,\pi[2]}, \dots, H_{1:M,\pi[m]}]$ representing the transfer functions of the other streams. The $\pi[1]$ -th stream is a result of linear combining of signals from M antennas 40 with weights that are defined by the entries of the projected vector. This type of interference cancellation, known as zero forcing, remains sub-optimal in the presence of noise. A better performance may be achieved with minimum mean square error (MMSE) cancellation (in MMSE canceller 44). This approach maximizes the output SINR. To apply MMSE extraction to the $\pi[1]$ -th stream, we compute $m \times M$ vector

$$W_{\pi[1]}^{(1)} = \left(H_{\pi[1]}^* \left(\sigma_s^2 H H^* + \sigma_n^2 I_M \right)^{-1} H_{\pi[1]} \right)^{-1} H_{\pi[1]}^* \left(\sigma_s^2 H H^* + \sigma_n^2 I_M \right)^{-1}. \quad (7)$$

Here the superscript (*) stands for a matrix conjugate transpose, I_M is the $M \times M$ identity matrix, σ_s^2 is the (average) power of every transmitted signal and σ_n^2 is the ambient noise power. The $\pi[1]$ -th stream is a result of linear combining of signals from M antennas 40 with weights that are defined by the respective entries of $W_{\pi[1]}^{(1)}$. Whenever the ambient noise at different antennas is uncorrelated, MMSE cancellation results in the highest possible SINR:

$$\text{SINR}_{\pi[1]}^{(1)} = \left[\left(H_{\pi[1]}^* \left(\sigma_s^2 H H^* + \sigma_n^2 I_M \right)^{-1} H_{\pi[1]} \right)^{-1} - 1 \right]^{-1}. \quad (8)$$

The extracted $\pi[1]$ -th stream 45 is forwarded by the MMSE canceller 44 to the decoder 46 which recovers the corresponding stream of user bits 47. These user bits 47 are encoded and modulated again by an encoder/modulator 48 to the sequence of channel symbols 49. The sequence of symbols 49 is scaled by the respective entries of the transfer function $H_{\pi[1]}$ (by means of the multipliers 56) to produce the contributions of the $\pi[1]$ -th stream to all M receiver branches. These contributions are extracted from the corresponding received signals by means of subtractors 60 as indicated in Figure 6. The resulting M signals 55 are free from the contributions of the $\pi[1]$ -th stream 45. The described procedure is applied

recursively so that at the n -th layer/stage, a $\pi[n]$ -th stream is extracted with certain $SINR_{\pi[n]}^{(n)}$, after the MMSE cancellation of the remaining $(n-1)$ interfering streams by means of filter $W_{\pi[n]}^{(n)}$, its contribution is reconstructed and removed from the received signals, $1 \leq n \leq m$ (except for the last layer/stage where the removal is not necessary). In Figure 6 also the second layer/stage and the m -th layer/stage are shown. This second stage is comprised of MMSE canceller 50, decoder 52, encoder/modulator 54, multipliers 64 and subtractors 68. The m -th stage is only comprised of MMSE canceller 74 (which comprises a Maximum Ratio Combiner (MRC) filter since no interference is present at this stage) and decoder 76. The receiver 16 further comprises a multiplexer 72 which multiplexes the decoded information subsignals 47 of the m layers/stages into an information signal 77 comprising user bits.

The throughput of this MIMO transmission system depends on the set of SINR values $SINR_{\pi[1]}^{(1)}, \dots, SINR_{\pi[m]}^{(m)}$. Hence, the order $\pi = \{\pi[n], 1 \leq n \leq m\}$ of the processing of m streams may be critical. To highlight the impact of the processing order on the system throughput, note that the symmetry of different subchannels and the absence of channel knowledge at the transmitter yield equal transmission rates (throughputs) to be used for all subchannels. The overall throughput of such a system equals to m times the throughput of one subchannel. Finally, the throughput per subchannel is limited by the minimum of their respective throughputs which is defined by $\min\{SINR_{\pi[1]}^{(1)}, \dots, SINR_{\pi[m]}^{(m)}\}$. The maximum throughput, therefore, corresponds to the maximum of $\min\{SINR_{\pi[1]}^{(1)}, \dots, SINR_{\pi[m]}^{(m)}\}$ whereas the optimal processing order is defined by such π that maximizes $\min\{SINR_{\pi[1]}^{(1)}, \dots, SINR_{\pi[m]}^{(m)}\}$. As shown in the above mentioned European Patent Application, the optimal processing order π is achieved when, at every stage, the subchannel is selected which maximizes the local SINR:

$$\pi[n] = \arg \max_k \left\{ SINR_k^{(n)} : 1 \leq k \leq m, k \neq \pi[p] : 1 \leq p < n \right\} \quad 1 \leq n \leq m. \quad (9)$$

A MIMO transmitter system as described earlier in this section, with the transmitter and receiver as in Figure 5 and Figure 6 respectively, MMSE cancellation filters specified by (7), decision statistics defined by (8) and processing order defined in (9), is regarded as the baseline system. Let us analyze a theoretically achievable throughput of such a system. We will assume a narrow-band (non-selective) Rayleigh fading channel with fully uncorrelated transmit/receive antennas. This means that the entries of the channel matrix are

statistically independent complex Gaussian variables with zero mean and variance $(1/2)$ per complex dimension. Let us first consider a system with two transmit/receive antennas:

$M = m = 2$. For this setup, the outage ratios $SINR_{\pi[1]}^{(1)}$ and $SINR_{\pi[2]}^{(2)}$ for both layers have been estimated from 100000 independent Monte-Carlo trials for a wide range of the overall SNR per receive antenna (i.e. ratio of the average total signal power from all transmit antennas to the noise power at any receive antenna). The empirical SINR values for the outage rates of 10% and 1% are plotted in Figure 9 (showing outage SINR per layer/stage versus the overall SNR per receive antenna for outage rates 10% (left frame) and 1% (right frame), 2 transmit antennas, 2 receive antennas). The corresponding outage values of capacities (maximum throughputs), computed according to the standard relationship

$$C = \log_2 (1 + SINR) \text{ [bits / channel use]} \quad (10)$$

are plotted in Figure 10 (showing outage throughput per layer/stage versus the overall SNR per receive antenna for outage rates 10% (left frame) and 1% (right frame), 2 transmit antennas, 2 receive antennas). One can see that at small and moderate SNR, the first (upper) layer has a bigger throughput. The advantage of the first layer over the second (lower) one also depends on the designed outage rate. Note that the maximum throughput of the first layer is nearly two times as big as the throughput of the second layer in some cases of practical interest. Namely, the region of SNR around 6-8dB, outage rate 10% and less may be relevant for cellular communications in interference-limited environments, e.g. CDMA.

As explained earlier, the throughput of each subchannel within the baseline system may not exceed the minimum of throughputs observed at different layers. Therefore, the maximum total throughput of the baseline system is twice the minimum of these throughputs. The dashed line in Figure 11 shows the overall throughput of the baseline (standard) system versus the overall SNR for the outage rates of 10% (left frame) and 1% (right frame) respectively.

At this point, we note that the overall throughput may be increased up to the sum of the throughputs C_1 and C_2 of the two layers in the region of SNR and outage rates where $C_1 \geq C_2$. Indeed, one may notice that the transmission system with a transmitter 12 as in Figure 5 and a receiver 16 as in Figure 6 is a particular case of the general transmission scheme in Figure 1 where the capacities $C_1 \dots C_m$ stand for the outage throughputs achievable at layers 1 through m respectively while the permutation π defines the processing order of

the transmitted streams. The set of outage throughputs $C_1 \dots C_m$ is defined by the statistical description of the assumed propagation environment (in our example, uncorrelated Rayleigh fading). Usually, these throughputs are measured off-line and may be assumed known to the transmitter 12 and to the receiver 16. The permutation π depends on the channel realization.

5 This permutation is defined at the receiver 16, subject to the estimated channel matrix, and therefore is unknown to the transmitter 12. Hence, the baseline transmission system falls under the general scheme as depicted by Figure 1 and therefore the general channel coding principle described above applies in this case. Let us design the encoder that has to operate in presence of uncorrelated Rayleigh fading at SNR of 8 dB and 10% outage rate. In this case,

10 the throughputs achievable at layers 1 and 2 are $C_1 \approx 1.27$ and $C_2 \approx 0.81$ user bits per channel use respectively, see Figure 10. The practically achievable throughputs of this layers are, therefore, upper-bounded by C_1 and C_2 . These upper bounds are never achieved in practice since a (small) fraction of the spectral efficiency has to be sacrificed in order to satisfy the QoS requirements in terms of error rates. This fraction depends on the desired

15 features of a FEC and QoS requirements. The definition of practical throughput relates to the FEC design which is not addressed specifically in this document. Therefore, we will assume in this example an ideal FEC so that the maximum throughputs are achievable. Suppose that a block of data to be transmitted makes use of $N = 100$ channel uses. This may correspond to e.g. a block of 100 symbols of certain alphabet serially sent over the channel. According to

20 the block-diagram of the MPC encoder (Figure 3), we have to partition the set of

$$N(C_1 + C_2) = 127 + 81 = 208 \text{ user bits into two subsets of sizes}$$

$$N(C_1 - C_2) = 127 - 81 = 46 \text{ and } N2C_2 = 2 \cdot 81 = 162. \text{ The first subset is encoded with } (2,1, \bullet)$$

MDS code. The generator matrix of a binary MDS code $(2,1,1)$ is given in (6). According to the generating matrix, this MDS code is a rate $(1/2)$ repetition code. Hence 46 user bits of

25 the first subset are simply duplicated and the resulting copies are fed to the output streams via the multiplexers 34 in Figure 3. The second subset of 162 bits is partitioned into two blocks of 81 bits. These blocks contribute to the two output streams through the same multiplexers 34. Each output stream contains 127 bits. These streams are encoded by channel encoders 24, modulated by modulators 26 and transmitted via different antennas 28 as shown in Figure

30 7. Here, the means for transmitting the encoded information subsignals 25 are formed by the modulators 26 and the antennas 28. At the receiver 16, the standard OSIC extraction of the data streams is performed, as described above. According to the processing order π

determined at the receiver 16 via (9), the stream $\pi[1]$ is extracted at the first (upper) layer. Since the outage throughput of this layer is C_1 , the corresponding $NC_1 \approx 127$ user bits are successfully decoded in the decoder 46. According to the block diagram in Figure 8, the values 47 (by bit values, we understand either binary/hard decisions or real-valued/soft decisions) of these bits are fed to the “input 1” of the MPC decoder 80. Inside the MPC decoder 80 (see Figure 4), these values 47 are partitioned into a set of 46 bit values corresponding to the $\pi[1]$ -th output of the $(2,1,1)$ MDS code and the remaining 81 bit values. Check that decoding of the $(2,1,1)$ MDS code consists of replicating 46 bit values of the $\pi[1]$ -th output to both outputs of the decoder. While the first replica 117 is collected by the multiplexer 86, the second replica 83 is sent to the output “link 1” of the MPC decoder 80. At the second layer, the stream $\pi[2]$ is extracted. The FEC decoder 52 of this stream 59 benefits from the knowledge of 46 bit values out of the total 127 bits fed via “link 1” of the MPC decoder 80, see Figure 8. The remaining $NC_2 = 81$ user bits may be successfully recovered since the throughput of the second layer is $C_2 \approx 0.81$. The whole stream 47 of 127 bit values of the second stream is sent to the “input 2” of the MPC decoder 80. According to Figure 4, a subset of 81 bit values at this stage is merged with the subset of 81 bit values from the previous decoding stage and sent to the multiplexer 86. The total number $46 + 81 + 81 = 208$ of bit values collected in the multiplexer 86 equals to the total number of the transmitted user bits. In the case of soft-decision decoding, the soft bit values are converted into hard decisions on the user bits.

Note that the proposed channel coding scheme results in the overall throughput of $(C_1 + C_2) \approx 2.08$ bits per channel use. This yields a 28% improvement as compared to the baseline system with the overall throughput of $2C_2 \approx 1.62$ bits per channel use. The throughputs of baseline system and the proposed modification of this latter are plotted in Figure 11 for various SNR and outage rates of 10% and 1%. The improvement of the modified transmission system over the baseline systems varies from 10% to 100% and even higher, at low and moderate SNR.

In the general case of M and m , the transmitter 12 of the modified baseline system is shown in Figure 7. This transmitter 12 makes use of MPC encoder 30 shown in Figure 3. The receiver 16 of the modified baseline system is presented in Figure 8. The modified receiver 16 differs from the receiver 16 of the baseline system (shown in Figure 6) by the MPC decoder 80 and related links with FEC decoders 46, 52, 76, 128 of different layers.

It is worthwhile to mention that the FEC structure should be adapted to take advantage of the knowledge of known bits. Although the design of specific FEC codes is beyond the scope of this invention, some promising options for the choice of FEC have been indicated above.

In the remainder, we give a slightly more complex example of MPC coding,
 5 for a MIMO transmission system with $M = m = 3$. Under the assumption of uncorrelated Rayleigh fading, we have computed the outage SINR achievable at different layers, the corresponding outage throughputs per layer and the outage overall throughputs of the baseline (standard) and modified systems, for a wide range of SNR and outage rates 10% and 1%, see Figures 12-14. As earlier, we select SNR of 8 dB, 10% outage rate and
 10 $N = 100$ channel uses per block. Figure 12 shows outage SINR per layer/stage versus the overall SNR per receive antenna for outage rates 10% (left frame) and 1% (right frame), 3 transmit antennas, 3 receive antennas. Figure 13 shows outage throughput per layer/stage versus the overall SNR per receive antenna for outage rates 10% (left frame) and 1% (right frame), 3 transmit antennas, 3 receive antennas. Figure 14 shows total outage throughput of
 15 the standard and the modified system versus the overall SNR per receive antenna for outage rates 10% (left frame) and 1% (right frame), 3 transmit antennas, 3 receive antennas.

First of all, we need to find the set of maximum throughputs $\underline{C}_1, \underline{C}_2, \underline{C}_3$ that deliver the maximum overall throughput C_x under the constraint (4). According to Figure 13, we have $C_1 \approx 1.51$, $C_2 \approx 1.33$, $C_3 \approx 0.95$. Note that (4) is satisfied with $\underline{C}_1 = C_1$, $\underline{C}_2 = C_2$
 20 and $\underline{C}_3 = C_3$. The transmitter 12 operates according to Figure 5 with MPC encoder 30 described in Figure 3. The total number of user bits is

$$N(C_1 + C_2 + C_3) = 151 + 133 + 95 = 379$$
. These 379 user bits are partitioned into $m = 3$ subsets with sizes $N(C_1 - C_2) = 18$, $N(C_2 - C_3) = 2 \cdot 38 = 76$ and $N C_3 = 3 \cdot 95 = 285$. The respective sizes of the output blocks are $N(C_1 - C_2) = 18$, $N(C_2 - C_3) = 38$ and
 25 $N C_3 = 95$, as described in Figure 3. Finally, the MDS codes used in the MPC encoder 30 are given by $(3,1,1)$ and $(3,2,1)$ codes with the respective generating matrices specified in (6).

The receiver 16 operates as described in Figure 8. The difference from the receiver 16 of the baseline system in Figure 6 is in the MPC decoder block 80 and its links to various decoding stages. This MPC decoder 80 operates similarly to the MPC decoder 80 in
 30 the previous example. It remains to specify the decoding of MDS codes. First, we note that the $(3,1,1)$ MDS code defined in (6) is rate $(1/3)$ repetition code. Hence the decoding procedure consists of copying the input of the decoder to its outputs. Let us consider the

decoding procedure for the $(3,2,1)$ code. First, we address hard decision decoding. As mentioned above, an exhaustive search can be used. In order to describe the search algorithm, we denote by b_1, b_2 a pair of inputs and by c_1, c_2, c_3 the respective outputs of the $(3,2,1)$ MDS code. According to the description of Figure 4, the task of the decoder is to find input bits b_1, b_2 and an output bit $c_{\pi[3]}$ given a pair of outputs $c_{\pi[1]}, c_{\pi[2]}$. For the $(3,2,1)$ MDS code defined in (6), the search is performed according to the lookup table shown in Figure 15 and Figure 16.

Finally, we consider soft decision decoding. Let us define I_1, I_2 a pair of real-valued metrics associated with the pair of input bits b_1, b_2 and O_1, O_2, O_3 three output metrics associated with the output bits c_1, c_2, c_3 . The task of the decoder is to compute input metrics I_1, I_2 and an output metric $O_{\pi[3]}$ given a pair of output metrics $O_{\pi[1]}, O_{\pi[2]}$. Conventionally, bit metrics are defined as log-likelihood ratios (LLR). The LLR values are usually computed via the maximum a posteriori probability (MAP) algorithm, or its simplified version widely known as Max-Log-MAP. The respective computation rules applicable to the $(3,2,1)$ MDS code in (6) are specified in Fig 17 and Figure 18 respectively.

It is worth noting that the exact expressions to compute LLR depend on the code structure and should be derived from the general rules of MAP or Max-Log-MAP computations for each particular code.

The scope of the invention is not limited to the embodiments explicitly disclosed. The invention is embodied in each new characteristic and each combination of characteristics. Any reference signs do not limit the scope of the claims. The word "comprising" does not exclude the presence of other elements or steps than those listed in a claim. Use of the word "a" or "an" preceding an element does not exclude the presence of a plurality of such elements.